

Klasifikasi Daging Sapi dan Daging Babi Menggunakan Convolutional Neural Network EfficientNet-B0 dengan Augmentasi Citra

Hafez Almirza, Jasril, Suwanto Sanjaya*, Lestari Handayani, Fadhilah Syafria

Fakultas Sains dan Teknologi, Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

Email: ¹hafezalmirza@gmail.com, ²jasril.uinsuska@gmail.com, ^{3,*}suwantosanjaya@uin-suska.ac.id,

⁴lestari.handayani@uin-suska.ac.id, ⁵fadhilah.syafria@uin-suska.ac.id

Email Penulis Korespondensi: suwantosanjaya@uin-suska.ac.id

Abstrak—Peningkatan penjualan daging sapi palsu sejalan dengan permintaan daging yang terus meningkat di Indonesia. Daging palsu, yaitu daging oplosan sapi dan babi maupun daging babi murni yang dijual sebagai daging sapi dapat dibedakan menggunakan klasifikasi citra. Penelitian ini melakukan klasifikasi daging babi, oplosan, dan sapi menggunakan model Convolutional Neural Network(CNN) arsitektur EfficientNet-B0. Penelitian ini menggunakan metode augmentasi citra untuk memperbanyak citra dengan tujuan meningkatkan akurasi klasifikasi. Total citra asli adalah 900 citra, sementara itu total citra hasil augmentasi adalah 9000. Data citra tersebut dibagi menggunakan dua rasio pembagian data, yaitu 80:20 dan 90:10. Hasil akurasi klasifikasi tertinggi diperoleh oleh model yang menggunakan citra augmentasi dan rasio pembagian data 90:10, dengan kombinasi hyperparameter optimizer Adamax, hidden activation Swish, dan learning rate 0.1 dengan hasil accuracy 97,11%, precision 97,14%, recall 97,11%, dan F1-Score 97,11%. Sementara itu, akurasi tertinggi model yang menggunakan citra asli dicapai oleh model yang menggunakan rasio pembagian 90:10 dengan kombinasi hyperparameter optimizer Adamax, hidden activation ReLU, dan learning rate 0.01 dengan hasil accuracy 96,78%, precision 96,92%, recall 96,78%, dan F1-Score 96,78%. Hasil penelitian menunjukkan bahwa penggunaan metode augmentasi citra dapat meningkatkan akurasi klasifikasi

Kata Kunci: Klasifikasi; Convolutional Neural Network; EfficientNet-B0; Citra; Augmentasi

Abstract—The increase in counterfeit beef sales is in line with the growing demand for meat in Indonesia. Counterfeit meat, namely mixed beef and pork and pure pork sold as beef, can be distinguished using image classification. This study classifies pork, mixed, and beef using the Convolutional Neural Network (CNN) model of the EfficientNet-B0 architecture. This study uses the image augmentation method to augment the image with the aim of improving classification accuracy. The total original image is 900, while the total augmented image is 9000. The image data is divided using two data division ratios, namely 80:20 and 90:10. The highest classification accuracy results were obtained by a model using augmented images and a data division ratio of 90:10, with a combination of Adamax hyperparameter optimizer, Swish hidden activation, and a learning rate of 0.1, with an accuracy of 97.11%, precision of 97.14%, recall of 97.11%, and F1-Score of 97.11%. Meanwhile, the highest accuracy of the model using the original image is achieved by the model using a 90:10 division ratio with a combination of hyperparameter optimizer Adamax, hidden activation ReLU, and learning rate 0.01 with the results of accuracy 96.78%, precision 96.92%, recall 96.78%, and F1-Score 96.78%. The results show that the use of image augmentation methods can improve classification accuracy.

Keywords: Classification; Convolutional Neural Network; EfficientNet-B0; Image; Augmentation

1. PENDAHULUAN

Mengacu pada data *Organization for Economic Cooperation and Development* (OECD), konsumsi daging sapi Indonesia tahun 2022 adalah 2.226 per kapita, meningkat dari tahun 2021 yang hanya 2.203 perkapita. Meningkatnya konsumsi daging sapi masyarakat menyebabkan kelangkaan. Kelangkaan daging sapi dan meningkatnya permintaan masyarakat terhadap daging sapi dimanfaatkan oleh pedagang yang tidak bertanggung jawab dengan mencampur daging babi pada daging sapi. Pada tahun 2016 di kota Surabaya, polisi menyita barang bukti 16 kg daging sapi bercampur daging babi[1]. Kasus serupa juga terjadi di kota Tangerang pada tahun 2020, polisi menangkap seorang pelaku yang mencampur daging sapi dengan daging babi[2]. Di tahun yang sama, Polresta Bandung menjangkir empat pelaku pengoplos daging babi yang dijual seolah-olah daging sapi di wilayah Kabupaten Bandung[3].

Daging sapi, babi dan oplosan dapat diklasifikasikan dengan memanfaatkan teknologi *image processing* melalui algoritma *Machine Learning*(ML). Beberapa penelitian terkait klasifikasi menggunakan ML telah beberapa kali dilakukan, seperti penelitian Budianita *et. al.*, dilakukan pengolahan citra untuk klasifikasi daging babi dan daging sapi berdasarkan warna dan tekstur dengan metode *Hue Saturation Value* (HSV) dan *Gray Level Co-Occurrence Matrix* (GLCM) lalu diklasifikasikan menggunakan algoritma K-Nearest *Neighbour*(KNN)[4]. Penelitian Handayani *et. al.*, melakukan klasifikasi citra tekstur daging babi dan sapi menggunakan *Probabilistik Neural Network*[5]. Penelitian lainnya dilakukan oleh Malikhah *et. al.*, menggunakan metode *Ensemble Learning*[6]. Pada saat ini, klasifikasi citra daging babi dan sapi menggunakan *machine learning* telah digantikan dengan metode *Deep Learning* dengan *Convolutional Neural Network*(CNN).

Penelitian terkait klasifikasi daging sapi, babi, dan oplosan menggunakan *CNN* telah dilakukan beberapa kali. Penelitian mengenai klasifikasi daging sapi dan babi dilakukan Laluma *et. al.*, menggunakan CNN dengan *classifier Multilayer Layer Perceptron* (MLP) [7]. Penelitian lainnya dilakukan oleh Artya *et. al.*, menggunakan CNN dengan arsitektur AlexNet, memberikan hasil akurasi klasifikasi 68,6% menggunakan *dataset Local Binary Pattern*(LBP) dan 84,1% menggunakan dataset asli[8]. Lasniari *et. al.*, melakukan penelitian terkait klasifikasi jenis daging menggunakan CNN arsitektur *ResNet-50* dengan tingkat akurasi 87,64% [9]. Alhafis *et. al.*, melakukan penelitian menggunakan CNN

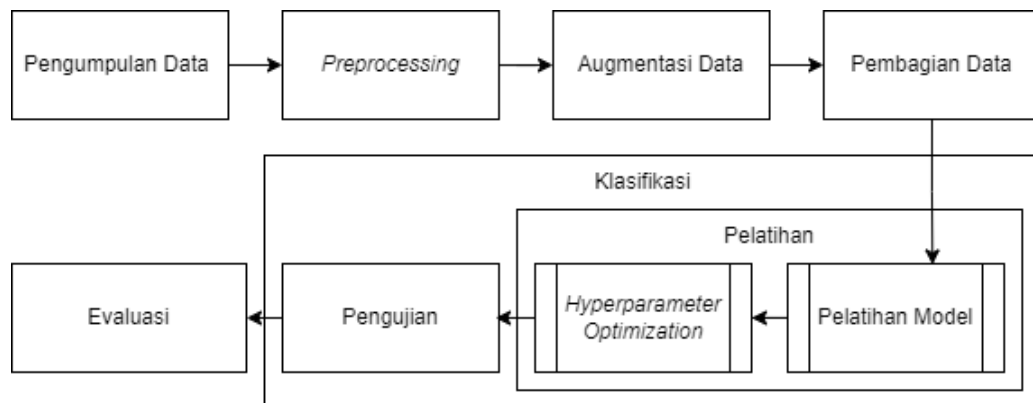
arsitektur EfficientNet-B0 dan metode *Contrast Limited Adaptive Histogram Equalization*(CLAHE) untuk meningkatkan kualitas citra dengan hasil terbaik diperoleh oleh citra tanpa metode pengolahan CLAHE[10].

Penelitian sebelumnya yang menggunakan CNN dilakukan oleh Bhupendra *et. al.*, tentang klasifikasi butir beras yang telah digiling menggunakan CNN arsitektur EfficientNet-B0, ResNet-50, InceptionV3, MobileNetV2, dan MobileNetV3 dengan hasil akurasi tertinggi dicapai oleh arsitektur EfficientNet-B0 yang memiliki nilai akurasi 98,37%[11]. Penelitian lain mengenai klasifikasi citra menggunakan CNN dilakukan Phiadelvira dengan teknik augmentasi citra untuk klasifikasi kanker serviks [12]. Penelitian tersebut memiliki nilai akurasi 100% menggunakan citra hasil augmentasi, sedangkan jika tanpa proses augmentasi hanya mendapatkan nilai tertinggi 66.67%.

Penelitian ini menggunakan CNN arsitektur EfficientNet-B0 dengan metode Augmentasi untuk meningkatkan jumlah citra. Penggunaan teknik augmentasi citra pada penelitian ini dilakukan untuk meningkatkan akurasi klasifikasi. Adapun tujuan penelitian ini adalah untuk melihat kinerja CNN arsitektur EfficientNet-B0 untuk klasifikasi daging sapi dan babi serta membandingkan hasil klasifikasi model yg menggunakan citra asli dan model dengan augmentasi citra.

2. METODOLOGI PENELITIAN

Tahapan penelitian merupakan bagian integral dalam proses penelitian yang dirancang secara sistematis untuk mencapai tujuan yang telah ditetapkan. Melalui tahapan-tahapan ini, peneliti dapat memperoleh pemahaman mendalam tentang subjek penelitian dan menghasilkan temuan yang valid dan bermakna. Setiap tahapan memiliki peran penting dalam menyusun kerangka penelitian, merancang metode yang tepat, mengumpulkan dan menganalisis data, serta menarik kesimpulan yang relevan. Tahapan penelitian yang akan diterapkan pada penelitian ini adalah sebagai berikut.



Gambar 1. Tahapan Penelitian

2.1 Pengumpulan Data

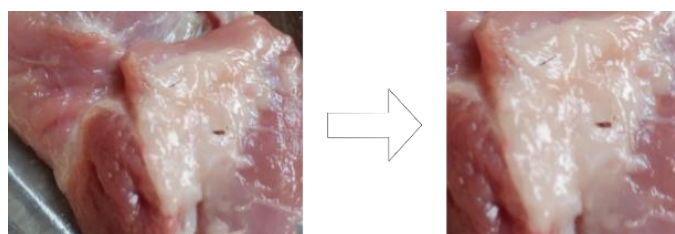
Tahap pengumpulan data dilakukan dengan mengambil citra daging babi, oplosan, dan sapi. Data citra yang digunakan pada penelitian ini adalah citra yang didapat secara mandiri. Daging didapatkan dari pasar yang berlokasi di Kota Pekanbaru, Riau. Citra setiap jenis daging diambil secara langsung menggunakan kamera ponsel Xiaomi Redmi Note 8 Pro dengan resolusi 64MP dan Vivo V20 dengan resolusi kamera 64MP. Data yang dikumpulkan berjumlah 300 untuk setiap jenis daging, sehingga totalnya adalah 900 citra daging.

2.2 Preprocessing

Citra daging yang telah diambil diolah pada tahap *preprocessing*. Tahap ini bertujuan untuk menyeragamkan citra daging sehingga proses ekstraksi informasi lebih akurat. Teknik yang dilakukan pada tahap ini adalah *crop* dan *resize*. Seluruh tahapan *preprocessing* dilakukan menggunakan bahasa pemrograman *python*.

a. Crop

Crop citra dilakukan untuk menghilangkan bagian yang tidak dibutuhkan pada tepi citra yang tidak memberikan informasi pola dari daging. Bagian yang tidak diinginkan atau biasa disebut *noise* dapat mengganggu proses ekstraksi informasi dari citra.



Gambar 2. *Crop* pada citra daging.

b. *Resize*

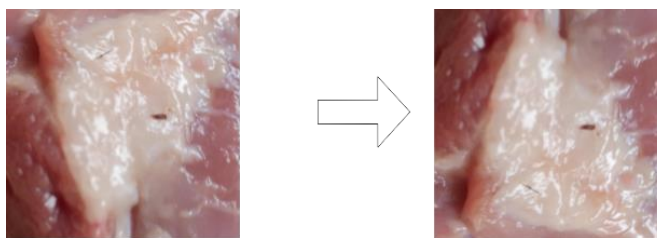
Resize adalah proses untuk menyeragamkan ukuran citra yang sebelumnya telah di-*crop*. Ukuran citra yang digunakan model penelitian ini adalah 224×224 piksel, yang mana merupakan ukuran yang ditetapkan pada arsitektur EfficientNet-B0.



Gambar 3. *Resize* pada citra daging.

2.3 **Augmentasi**

Data citra yang digunakan pada penelitian ini untuk klasifikasi dikategorikan menjadi data asli dan data augmentasi. Hal ini dilakukan untuk membandingkan kualitas model yang dilatih antara 2 data tersebut. Augmentasi citra bertujuan untuk memperbanyak jumlah dataset sehingga model yang dilatih akan lebih maksimal tanpa harus melakukan sampel tambahan data[9]. Pada penelitian ini teknik augmentasi yang dilakukan adalah *rotation*, *horizontal flip*, *vertical flip*, dan *brightness shift*. Jumlah yang dihasilkan dari tahap augmentasi untuk tiap citra adalah 10 citra augmentasi, sehingga total seluruh citra augmentasi untuk adalah 3000 citra untuk masing-masing daging.



Gambar 4. Augmentasi citra dengan *flip vertical*

2.4 **Pembagian Citra**

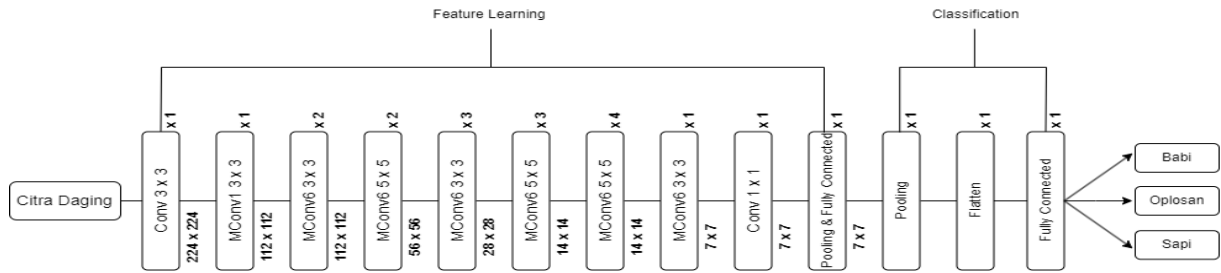
Tahap ini dilakukan untuk membagi citra asli dan citra yang telah di augmentasi menjadi 3 jenis data, yaitu data latih, data validasi, dan data uji. Data dibagi dengan rasio 80:20 dan 90:10. Pembagian data dilakukan 2 tahap, yaitu data dibagi menjadi. Berikut ini adalah tabel pembagian jumlah data :

Tabel 1. Pembagian data citra

| Rasio | Data | Kelas | Jumlah citra | | |
|---------|------------|---------|----------------|---------------|----------------|
| | | | Data pelatihan | | Data pengujian |
| | | | Data latih | Data validasi | Data uji |
| 80 : 20 | Asli | Babi | 192 | 48 | 60 |
| | | Oplosan | 192 | 48 | 60 |
| | | Sapi | 192 | 48 | 60 |
| | Augmentasi | Babi | 1920 | 480 | 600 |
| | | Oplosan | 1920 | 480 | 600 |
| | | Sapi | 1920 | 480 | 600 |
| 90 : 10 | Asli | Babi | 243 | 27 | 30 |
| | | Oplosan | 243 | 27 | 30 |
| | | Sapi | 243 | 27 | 30 |
| | Augmentasi | Babi | 2430 | 270 | 300 |
| | | Oplosan | 2430 | 270 | 300 |
| | | Sapi | 2430 | 270 | 300 |

2.5 **Klasifikasi**

Klasifikasi gambar pada penelitian ini dilakukan dengan model CNN arsitektur Efficient-NetB0. CNN merupakan arsitektur jaringan operasi konvolusi yang terdiri dari lapisan *convolutional*, *pooling*, dan *fully connected* yang dapat mempelajari fitur pada citra [13]. CNN mengaplikasi *filter* konvolusi dua dimensi atau lebih pada lapisan input [14]. EfficientNet-B0 adalah salah satu arsitektur *pre-trained* model yang mampu memberikan akurasi yang tinggi memperbaiki efektivitas model dengan pengoptimalan parameter[15]. Gambar 5. Arsitektur EfficientNet-B0 Gambar 5 adalah ilustrasi arsitektur EfficientNet-B0 yang digunakan pada penelitian ini.



Gambar 5. Arsitektur EfficientNet-B0

a. Pelatihan

Model EfficientNet-B0 dilatih menggunakan data latih yang telah dibagi sebelumnya. Jumlah data yang digunakan adalah dapat dilihat pada Tabel 1. Pada waktu pelatihan, data validasi digunakan untuk mengevaluasi model yang sedang dilatih untuk mengurangi resiko terjadinya *overfitting*.

Pada tahap pelatihan, diterapkan proses *hyperparameter optimization* atau optimasi *hyperparameter*. Optimasi *hyperparameter* adalah upaya untuk mengidentifikasi kombinasi *hyperparameter* dan mengoptimalkan loss function pada model *neural network*[16]. Pada penelitian ini optimasi *hyperparameter* pada diterapkan pada *activation function*, *optimizer*, dan *learning rate*.

1. *Activation function* atau fungsi aktivasi adalah fungsi yang menentukan aktif atau tidaknya neuron pada lapisan *neural network* dengan tujuan untuk menciptakan ke-nirlanjar-an pada output neuron. Fungsi aktivasi yang akan digunakan pada penelitian ini adalah Leaky ReLU, ReLU, dan Swish.
2. *Optimizer* yang akan digunakan pada penelitian ini adalah RAdam(Rectified Adaptive Moment Estimation) dan Adamax. RAdam adalah variasi dari adam yang menawarkan pengkoreksian dari variasi adaptive learning rate. Berdasarkan Valova (2020), RAdam memberikan performa lebih baik dari pada Adam[17].
3. *Learning rate* yang akan diterapkan dalam penelitian ini adalah 0.01 dan 0.001.

b. Pengujian

Tahapan ini adalah tahapan menguji kemampuan klasifikasi model yang telah dilatih. Tahap ini dilakukan menggunakan data uji yang berjumlah masing-masing 60 citra asli dan 600 untuk citra augmentasi pada rasio pembagian 80:20 serta 30 citra asli dan 300 citra augmentasi pada rasio pembagian 90:10.

2.6 Evaluasi

Evaluasi adalah tahap pemeriksaan akurasi hasil eksperimen skenario pengujian dan optimasi *hyperparameter*[9]. Hasil eksperimen akan dianalisis untuk diambil kesimpulan sebagai dasar kombinasi *hyperparameter* pada percobaan selanjutnya untuk menemukan kombinasi dengan performa terbaik. Pada penelitian ini evaluasi dilakukan evaluasi menggunakan *confusion matrix*. *Confusion Matrix* adalah matriks evaluasi yang membandingkan hasil klasifikasi dengan nilai sebenarnya[18]. Model dievaluasi dengan metrik *accuracy*, *precision*, *recall*, dan *F1-score*.

1. *Accuracy* adalah metrik untuk model klasifikasi yang mengukur perbandingan antara prediksi benar dengan keseluruhan citra.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{1}$$

2. *Precision* adalah rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif.

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

3. *Recall* adalah rasio prediksi benar dibandingkan dengan keseluruhan data yang benar positif.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{3}$$

4. *F1-Score* merupakan kombinasi antara *precision* dan *recall* dalam satu metrik. *F1-Score* menghitung rata-rata dari *precision* dan *recall*.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{4}$$

3. HASIL DAN PEMBAHASAN

3.1 Pelatihan

Pada penelitian ini dilakukan eksperimen pada lingkungan *Google Colab* dengan bahasa pemrograman pemrograman *python*. Penelitian ini memanfaatkan *library tensorflow, numpy, pandas, matplotlib*, dan *efficientnet* untuk setiap eksperimen. Setiap eksperimen dilakukan dengan iterasi(*epoch*) yang sama, yaitu 25 kali. Penelitian ini menggunakan

CNN metode *transfer learning* EfficientNet-B0 yang diperoleh dari *library efficientnet*. Model klasifikasi dibangun diatas EfficientNet-B0 dengan menargetkan 3 kelas, yaitu : sapi, oplosan, dan babi.

Dalam upaya mencari model dengan kinerja terbaik, dilakukan beberapa eksperimen dengan kombinasi *hyperparameter* yang berbeda berdasarkan arsitektur CNN EfficientNet-B0. Nilai akurasi model pada tahap penelitian dapat dilihat pada Tabel 2.

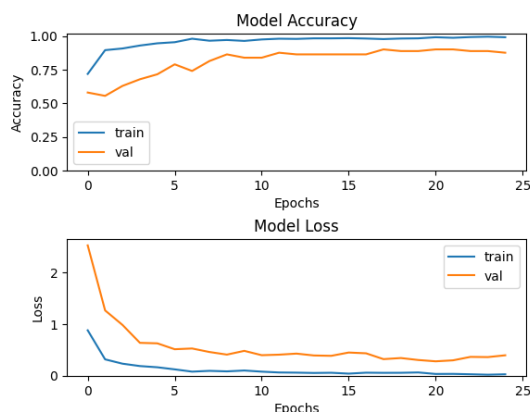
Tabel 2. Hasil pelatihan model Efficientnet-B0

| No. Model | Rasio | Dataset | Optimizer | Hidden Activation | LR | Waktu training | Accuracy | Validation Accuracy | | |
|-----------|------------|---------|------------|-------------------|--------|----------------|---------------|---------------------|---------------|---------------|
| 1 | 80:20 | Asli | Adamax | Leaky ReLU | 0.01 | 30m 30s | 99,13% | 50,69% | | |
| 2 | | | Adamax | Leaky ReLU | 0.1 | 30m 30s | 97,74% | 59,03% | | |
| 3 | | | Adamax | ReLU | 0.01 | 29m 29s | 99,13% | 61,81% | | |
| 4 | | | Adamax | ReLU | 0.1 | 31m 31s | 96,35% | 54,86% | | |
| 5 | | | Adamax | Swish | 0.01 | 29m 29s | 99,48% | 61,11% | | |
| 6 | | | Adamax | Swish | 0.1 | 30m 30s | 97,05% | 65,28% | | |
| 7 | | | RAdam | Leaky ReLU | 0.01 | 30m 30s | 97,22% | 55,56% | | |
| 8 | | | RAdam | Leaky ReLU | 0.1 | 30m 30s | 94,27% | 54,17% | | |
| 9 | | | RAdam | ReLU | 0.01 | 30m 30s | 98,44% | 55,56% | | |
| 10 | | | RAdam | ReLU | 0.1 | 30m 30s | 93,75% | 43,75% | | |
| 11 | | | RAdam | Swish | 0.01 | 27m 27s | 99,13% | 61,81% | | |
| 12 | | | RAdam | Swish | 0.1 | 29m 29s | 93,75% | 59,03% | | |
| 13 | Augmentasi | Asli | Adamax | Leaky ReLU | 0.01 | 4h 53m 53s | 98,56% | 72,78% | | |
| 14 | | | Adamax | Leaky ReLU | 0.1 | 4h 9m 9s | 96,68% | 70,49% | | |
| 15 | | | Adamax | ReLU | 0.01 | 4h 36m 36s | 98,66% | 74,10% | | |
| 16 | | | Adamax | ReLU | 0.1 | 4h 44m 44s | 97,14% | 74,37% | | |
| 17 | | | Adamax | Swish | 0.01 | 4h 38m 38s | 98,84% | 73,40% | | |
| 18 | | | Adamax | Swish | 0.1 | 3h 43m 43s | 97,93% | 77,99% | | |
| 19 | | | RAdam | Leaky ReLU | 0.01 | 4h 23m 23s | 97,27% | 67,64% | | |
| 20 | | | RAdam | Leaky ReLU | 0.1 | 4h 31m 31s | 89,24% | 77,08% | | |
| 21 | | | RAdam | ReLU | 0.01 | 4h 48m 48s | 97,38% | 77,92% | | |
| 22 | | | RAdam | ReLU | 0.1 | 4h 29m 29s | 91,77% | 70,69% | | |
| 23 | | | RAdam | Swish | 0.01 | 4h 10m 10s | 97,52% | 64,51% | | |
| 24 | | | RAdam | Swish | 0.1 | 4h 28m 28s | 92,19% | 65,97% | | |
| 25 | 90:10 | Asli | Adamax | Leaky ReLU | 0.01 | 30m 46s | 98,90% | 85,19% | | |
| 26 | | | Adamax | Leaky ReLU | 0.1 | 36m 29s | 97,67% | 83,95% | | |
| 27 | | | Adamax | ReLU | 0.01 | 33m 25s | 98,90% | 81,48% | | |
| 28 | | | Adamax | ReLU | 0.1 | 33m 36s | 98,08% | 83,95% | | |
| 29 | | | Adamax | Swish | 0.01 | 32m 06s | 99,18% | 87,65% | | |
| 30 | | | Adamax | Swish | 0.1 | 35m 49s | 97,39% | 82,72% | | |
| 31 | | | RAdam | Leaky ReLU | 0.01 | 34m 13s | 98,77% | 80,25% | | |
| 32 | | | RAdam | Leaky ReLU | 0.1 | 34m 11s | 91,77% | 79,01% | | |
| 33 | | | RAdam | ReLU | 0.01 | 37m 28s | 97,26% | 87,65% | | |
| 34 | | | RAdam | ReLU | 0.1 | 33m 53s | 92,32% | 80,25% | | |
| 35 | | | RAdam | Swish | 0.01 | 33m 25s | 98,35% | 77,78% | | |
| 36 | | | RAdam | Swish | 0.1 | 37m 31s | 92,73% | 79,01% | | |
| 37 | | | Augmentasi | Asli | Adamax | Leaky ReLU | 0.01 | 4h 38m 39s | 98,67% | 70,99% |
| 38 | | | | | Adamax | Leaky ReLU | 0.1 | 5h 49m 55s | 96,38% | 68,64% |
| 39 | | | | | Adamax | ReLU | 0.01 | 5h 07m 35s | 98,79% | 68,15% |
| 40 | | | | | Adamax | ReLU | 0.1 | 5h 23m 48s | 97,37% | 67,65% |
| 41 | | | | | Adamax | Swish | 0.01 | 3h 30m 03s | 99,05% | 73,95% |
| 42 | | | | | Adamax | Swish | 0.1 | 5h 29m 09s | 97,17% | 68,02% |
| 43 | | | | | RAdam | Leaky ReLU | 0.01 | 5h 21m 53s | 97,05% | 71,11% |
| 44 | | | | | RAdam | Leaky ReLU | 0.1 | 5h 33m 59s | 88,41% | 66,91% |
| 45 | | | | | RAdam | ReLU | 0.01 | 5h 03m 28s | 97,78% | 70,12% |
| 46 | | | | | RAdam | ReLU | 0.1 | 5h 15m 31s | 87,94% | 69,01% |
| 47 | | | | | RAdam | Swish | 0.01 | 4h 40m 43s | 97,54% | 70,00% |
| 48 | | | | | RAdam | Swish | 0.1 | 5h 17m 40s | 91,33% | 74,81% |

LR = Learning Rate

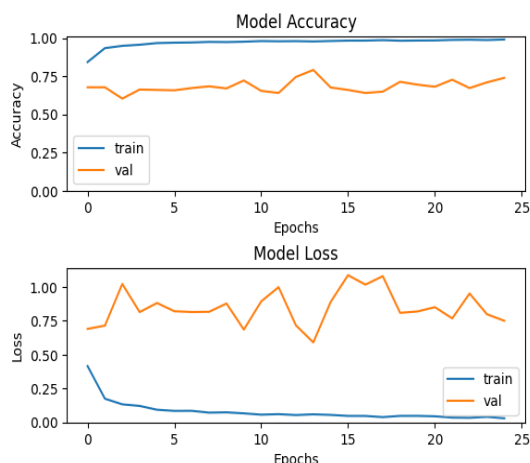
Pada Tabel 2 dapat dilihat bahwa rasio pembagian 90:10 memberikan nilai akurasi validasi yang lebih baik daripada rasio 80:20. Akurasi pelatihan tertinggi dicapai oleh model percobaan ke-29 pada dataset citra asli. Sementara

itu, untuk dataset augmentasi dicapai oleh pada percobaan ke-41. Grafik *loss* dan *accuracy* dari pelatihan 2 model tersebut dapat dilihat pada .



Gambar 6. Grafik *accuracy* dan *loss* model 29 yang menggunakan citra asli

Gambar 6 merupakan grafik *loss* dari model 29 dengan *optimizer* Adamax, *activation* Swish, dan *learning rate* 0.01 menggunakan dataset asli. Tingkat akurasi pelatihan dan akurasi validasi adalah 99,18% dan 87,65%.



Gambar 7. Grafik *accuracy* dan *loss* model ke-41 yang menggunakan citra augmentasi

Gambar 7 merupakan grafik *loss* dari model *optimizer* Adamax, *activation* Swish, dan *learning rate* 0.01 menggunakan dataset augmentasi. Angka akhir akurasi pelatihan dan akurasi validasi adalah 99,05% dan 73,95%.

3.2 Pengujian

Model-model yang telah dilatih akan diuji untuk klasifikasi citra daging menggunakan data uji. Hasil pengujian model yang memiliki performa terbaik dalam klasifikasi dapat dilihat pada Tabel 3.

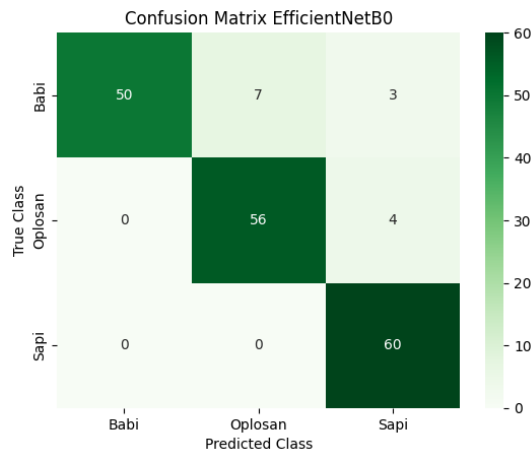
Tabel 3. Hasil pengujian menggunakan data uji

| No. Model | Rasio | Dataset | Optimizer | Hidden act. | LR | Accuracy | Precision | Recall | F1-Score | Presisi Sapi |
|-----------|-------|------------|-----------|-------------|------|----------|-----------|--------|----------|--------------|
| 2 | 80:20 | Asli | Adamax | Leaky ReLU | 0.1 | 91,67% | 92,10% | 91,67% | 91,66% | 92,06% |
| 3 | 80:20 | Asli | Adamax | ReLU | 0.01 | 92,22% | 92,81% | 92,22% | 92,15% | 89,55% |
| 20 | 80:20 | Augmentasi | RAdam | Leaky ReLU | 0.1 | 94,17% | 94,17% | 94,17% | 94,17% | 94,47% |
| 21 | 80:20 | Augmentasi | RAdam | ReLU | 0.01 | 95,06% | 95,12% | 95,06% | 95,06% | 92,50% |
| 27 | 90:10 | Asli | Adamax | ReLU | 0.01 | 96,78% | 96,92% | 96,78% | 96,78% | 93,75% |
| 29 | 90:10 | Asli | Adamax | Swish | 0.01 | 96,56% | 96,62% | 96,56% | 96,54% | 96,55% |
| 42 | 90:10 | Augmentasi | Adamax | Swish | 0.1 | 97,11% | 97,14% | 97,11% | 97,11% | 95,16% |
| 48 | 90:10 | Augmentasi | RAdam | Swish | 0.1 | 96,00% | 96,01% | 96,00% | 96,00% | 95,95% |

LR = Learning Rate

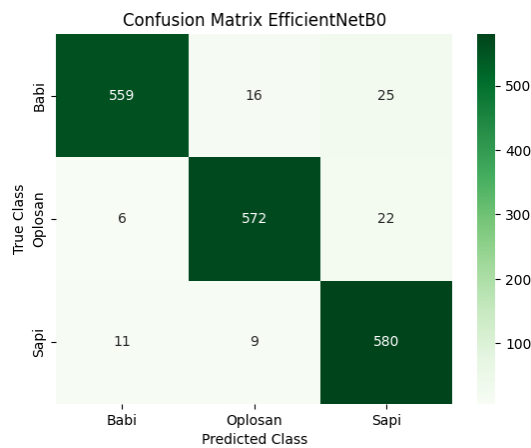
Tabel 3 merupakan tabel yang menunjukkan hasil pengujian klasifikasi menggunakan data uji pada beberapa model dengan kinerja terbaik. Model dengan rasio pembagian data 90:10 memberikan hasil yang lebih tinggi daripada model dengan rasio pembagian 80:20. Nilai tertinggi menggunakan rasio bagi 80:20 dicapai oleh model ke-3 menggunakan citra asli dan model ke-21 menggunakan citra augmentasi. Sedangkan untuk rasio bagi 90:10 dicapai oleh

model ke-27 menggunakan citra asli dan model ke-42 menggunakan citra augmentasi. Sementara itu angka presisi tertinggi untuk klasifikasi sapi dicapai oleh model ke-29 untuk citra asli serta model ke-48 untuk citra augmentasi. Nilai presisi prediksi terhadap dipilih untuk mengukur kualitas model dengan *false positive* terendah yang merupakan resiko minimum. Pengukuran evaluasi dihitung berdasarkan angka hasil klasifikasi menggunakan data uji yang dibentuk menjadi *confusion matrix*. Berikut ini adalah *confusion matrix* dari beberapa model diatas.



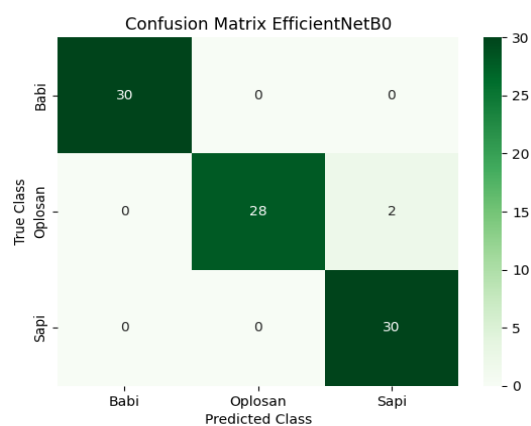
Gambar 8. *Confusion matrix* model ke-3

Gambar 8 merupakan *confusion matrix* dari model ke-3 yang menggunakan rasio bagi 80:20 dan citra asli, dengan kombinasi *hyperparameter* berikut: *optimizer* Adamax, *hidden activation* ReLU, dan *learning rate* 0,01. Model ini memiliki nilai *accuracy* 92,22%, *precision* 92,81%, *recall* 92,22%, dan *f1-score* 92,15%



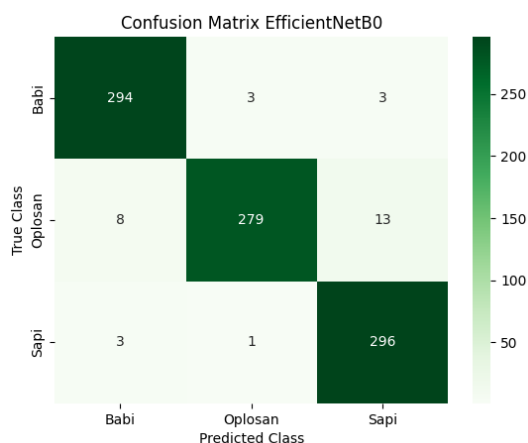
Gambar 9. *Confusion matrix* model ke-21

Gambar 9 merupakan *confusion matrix* dari model ke-21 yang menggunakan rasio bagi 80:20 dan citra augmentasi, dengan kombinasi *hyperparameter* berikut: *optimizer* RAdam, *hidden activation* ReLU, dan *learning rate* 0,01. Model ini memiliki nilai *accuracy* 95,06%, *precision* 95,12%, *recall* 95,06%, dan *f1-score* 95,06%



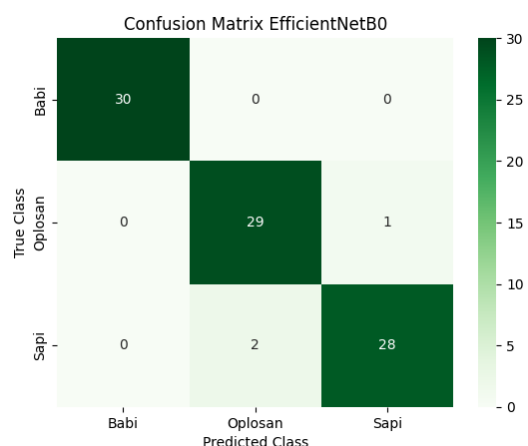
Gambar 10. *Confusion matrix* model ke-27

Gambar 10 menunjukkan *confusion matrix* pada hasil klasifikasi dari model ke-27 dengan *optimizer* Adamax, *activation* ReLU, dan *learning rate* 0.01 menggunakan dataset asli. Data yang digunakan pada pelatihan model ini menggunakan rasio bagi 90:10. Model ini memiliki nilai *accuracy* 96,78%, *precision* 96,92%, *recall* 96,78%, dan *f1-score* 96,78%



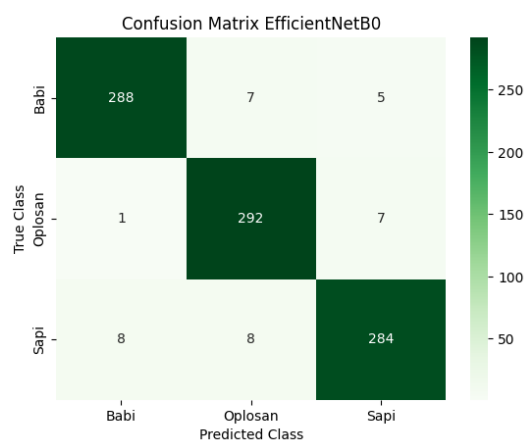
Gambar 11. *Confusion matrix* model ke-42

Gambar 11 adalah *Confusion Matrix* yang memetakan hasil klasifikasi dari model ke-42 dengan *optimizer* Adamax, *activation* Swish, dan *learning rate* 0.1 menggunakan dataset augmentasi. Model ini merupakan model dengan nilai akurasi tertinggi, dengan nilai *accuracy* 97,11%, *precision* 97,14%, *recall* 97,11%, dan *f1-score* 97,11%



Gambar 12. *Confusion matrix* model ke-29

Gambar 12 adalah *confusion matrix* yang memetakan hasil klasifikasi dari model ke-29 yang menggunakan rasio bagi 90:10. Model ke-29 memiliki kombinasi *hyperparameter* sebagai berikut : *optimizer* Adamax, *activation* Swish, dan *learning rate* 0.01 menggunakan dataset asli. Nilai presisi sapi pada model ini adalah 96,55%.



Gambar 13. *Confusion matrix* model ke-48

Gambar 13 menunjukkan confusion matrix dari model ke-48 yang menggunakan *optimizer* RAdam, *activation* Swish, dan *learning rate* 0.1 menggunakan dataset augmentasi. Nilai presisi sapi pada model ini adalah 95,95%.

4. KESIMPULAN

Klasifikasi citra daging sapi, daging babi, dan daging hasil oplosan pada penelitian ini menggunakan CNN arsitektur dasar EfficientNet-B0. Citra daging diperoleh secara mandiri dengan jumlah 300 citra setiap jenis daging. Data citra daging dibagi dengan rasio 80:20 dan 90:10. Data citra dibedakan menjadi data asli dengan jumlah 300 setiap jenis daging dan data hasil augmentasi dengan jumlah 3000 untuk tiap jenis daging. Pengujian pada model yang telah dilatih dilakukan dengan pengukuran accuracy, precision, recall, dan F1-Score. Hasil penelitian yang telah dilakukan menunjukkan hasil akurasi tertinggi yang menggunakan rasio bagi 90:10 dicapai oleh model menggunakan citra asli dengan *optimizer* Adamax, *hidden activation* Swish, dan *learning rate* 0,1 dengan hasil akurasi 97,11%, presisi 97,14%, *recall* 97,11%, dan *F1-Score* 97,11%. Sedangkan untuk rasio bagi 80:20, hasil akurasi tertinggi dicapai oleh model menggunakan citra augmentasi dengan *optimizer* RAdam, *hidden activation* ReLU, dan *learning rate* 0,01 dengan hasil akurasi 95,06%, presisi 95,12%, *recall* 95,06%, dan *F1-Score* 95,12%. Dapat disimpulkan bahwa penggunaan metode augmentasi citra dapat meningkatkan akurasi klasifikasi dan penggunaan rasio pembagian data 90:10 pada model memberikan performa lebih baik daripada rasio pembagian data 80:10. Untuk penelitian selanjutnya, direkomendasikan untuk penggunaan citra yang lebih beragam dan penggunaan teknik *learning rate scheduling* untuk meningkatkan akurasi, karena *learning rate* berpengaruh cukup signifikan pada penelitian ini.

REFERENCES

- [1] I. Wahyudiyanta, "Polisi Bongkar Penjualan Daging Sapi Dicampur Daging Babi di Surabaya," 2016. <https://news.detik.com/berita-jawa-timur/d-3218411/polisi-bongkar-penjualan-daging-sapi-dicampur-daging-babi-di-surabaya> (accessed Dec. 07, 2022).
- [2] S. Wiryono, "Fakta Kasus Daging Oplosan Sapi dan Babi di Tangerang, Kelabui Pembeli dengan Harga Murah Halaman all - Kompas.com," 2020. <https://megapolitan.kompas.com/read/2020/05/19/07120811/fakta-kasus-daging-oplosan-sapi-dan-babi-di-tangerang-kelabui-pembeli?page=all> (accessed Dec. 07, 2022).
- [3] M. Iqbal, "Polisi Sebut Babi Disulap Jadi Daging 'Sapi' Beredar di 3 Kecamatan Bandung," 2020. <https://news.detik.com/berita-jawa-barat/d-5010943/polisi-sebut-babi-disulap-jadi-daging-sapi-beredar-di-3-kecamatan-bandung> (accessed May 23, 2023).
- [4] E. Budianita, J. Jasril, and L. Handayani, "Implementasi Pengolahan Citra dan Klasifikasi K-Nearest Neighbour Untuk Membangun Aplikasi Pembeda Daging Sapi dan Babi Berbasis Web," *J. Sains dan Teknol. Ind.*, vol. 12, no. Vol 12, No 2 (2015): Juni 2015, pp. 242–247, 2015, [Online]. Available: <http://ejournal.uin-suska.ac.id/index.php/sitekin/article/view/1005>
- [5] L. Handayani *et al.*, "Comparison of target Probabilistic Neural network (PNN) classification for beef and pork," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 12, pp. 2753–2760, 2017.
- [6] M. Malikhah, R. Sarno, and S. I. Sabilla, "Ensemble Learning for Optimizing Classification of Pork Adulteration in Beef Based on Electronic Nose Dataset," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 4, pp. 44–55, 2021, doi: 10.22266/ijies2021.0831.05.
- [7] R. H. Laluma, B. Sugianto, A. Santriyana, A. G. Azwar, N. Nurwathi, and G. Gunawan, "Klasifikasi Perbedaan Daging Sapi Dan Daging Babi Dengan Metode Convolutional Neural Network Berbasis Web," *Infotronik J. Teknol. Inf. dan Elektron.*, vol. 6, no. 1, p. 1, 2021, doi: 10.32897/infotronik.2021.6.1.603.
- [8] A. H. Artya, J. Jasril, S. Sanjaya, F. Syafria, and E. Budianita, "Implementasi Convolutional Neural Network Untuk Klasifikasi Daging Menggunakan Fitur Ekstraksi Tekstur dan Arsitektur AlexNet," *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 3, p. 635, 2022, doi: 10.30865/jurikom.v9i3.4177.
- [9] S. Lasniari, J. Jasril, S. Sanjaya, F. Yanto, and M. Affandes, "Klasifikasi Citra Daging Babi dan Daging Sapi Menggunakan Deep Learning Arsitektur ResNet-50 dengan Augmentasi Citra," *J. Sist. Komput. dan Inform.*, vol. 3, no. 4, p. 450, 2022, doi: 10.30865/json.v3i4.4167.
- [10] G. Y. Alhafis, S. Sanjaya, F. Syafria, and E. Budianita, "Klasifikasi Citra Daging Sapi dan Daging Babi Menggunakan Ekstraksi Ciri dan Convolutional Neural Network," *J. Ris. Komputer*, vol. 9, no. 3, pp. 2407–389, 2022, doi: 10.30865/jurikom.v9i3.4175.
- [11] Bhupendra, K. Moses, A. Miglani, and P. Kumar Kankar, "Deep CNN-based damage classification of milled rice grains using a high-magnification image dataset," *Comput. Electron. Agric.*, vol. 195, p. 106811, Apr. 2022, doi: 10.1016/J.COMPAG.2022.106811.
- [12] B. Y. Phiadelvira, "Klasifikasi Kanker Serviks Berdasarkan Citra Kolposkopi Menggunakan Convolutional Neural Network (CNN) Model Alexnet," 2021, doi: 10.33387/jiko.v4i1.2606.
- [13] F. Hu, G. S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sens.*, vol. 7, no. 11, pp. 14680–14707, 2015, doi: 10.3390/rs71114680.
- [14] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A Guide to Convolutional Neural Networks for Computer Vision," *Synth. Lect. Comput. Vis.*, vol. 8, no. 1, pp. 1–207, 2018, doi: 10.2200/s00822ed1v01y201712cov015.
- [15] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019, doi: <https://doi.org/10.48550/arXiv.1905.11946>.
- [16] N. M. Aszemi and P. D. D. Dominic, "Hyperparameter optimization in convolutional neural network using genetic algorithms," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 6, pp. 269–278, 2019, doi: 10.14569/ijacsa.2019.0100638.
- [17] I. Valova, C. Harris, T. Mai, and N. Gueorgieva, "Optimization of convolutional neural networks for imbalanced set classification," *Procedia Comput. Sci.*, vol. 176, pp. 660–669, 2020, doi: 10.1016/j.procs.2020.09.038.
- [18] H. Aghdam and E. Heravi, *Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification*. 2017. doi: 10.1007/978-3-319-57550-6.